Signature
실습

OPENSSL SHA 링크

https://github.com/openssl/openssl/blob/master/include/openssl/sha.h

https://www.openssl.org/docs/man1.0.2/man3/SHA256.html

```
# define SHA256_DIGEST_LENGTH 32

typedef struct SHA256state_st {
    SHA_LONG h[8];
    SHA_LONG Nl, Nh;
    SHA_LONG data[SHA_LBLOCK];
    unsigned int num, md_len;
} SHA256_CTX;


//H(M)
unsigned char *SHA256(const unsigned char *d, size_t n, unsigned char *md);
//H(M1||M2||....)
int SHA256_Init(SHA256_CTX *c);
int SHA256_Update(SHA256_CTX *c, const void *data, size_t len);
int SHA256_Final(unsigned char *md, SHA256_CTX *c);
```

예제

```
#include <openssl/sha.h>
```

<H(M)>

```
U8 digest[SHA256_DIGEST_LENGTH]={0};
SHA256(M, strlen(M), digest);
```

hs, strlen()을 제외한 모든 변수들은 unsigned char * 로 넣어주면 된다.

<H(M||R)>

```
SHA256_CTX hs = {0};
SHA256_Init(&hs);
U8 digest[SHA256_DIGEST_LENGTH] = {0};
SHA256_Update(&hs, M, strlen(M));
SHA256_Update(&hs, R, BN_num_bytes(R));
SHA256_Final(digest, &hs);
```

RSA Signature

$\text{Keygen}(\lambda = 1024)$

$generate\ random\ 1024-bit\ prime\ p, q$

$N = pq,\ \phi(N) = (p-1)(q-1),$

$e \leftarrow Z_N\ where\ \gcd\big(e, \phi(N)\big) = 1,$

$d = e^{-1}\ mod\ \phi(N)$

$vk = (N, e),\ sk =$ (d)

$\text{Sign}(m, sk) : h = H(m),\ \sigma = h^d\ mod\ N$

$return\ \sigma$

$\text{Verify}(\sigma, m, vk) : h' = \sigma^e\ mod\ N$

$if\ h' == H(m)\ return\ 1$

$else\ return\ 0$

Schnorr Signature

Keygen($\lambda = 1024$)

$generate\ random\ 1024 - bit\ prime\ p$

g = generator, x←$Z_p, y = g^x\ mod\ p$

$vk = (p, g, y),\ \ sk = x$

Sign(m, sk) : $r \leftarrow Z_p,\ \ R = g^r\ mod\ p$,

$c = H(m||R),\ z = (r + xc) mod\ p - 1^{|Z_p^*|}$

return $\sigma = (c, z)$

Verify(σ, m, vk) : $R' = \frac{g^z}{y^c}\ mod\ p, c' = H(m||R')$

$if\ c == c'\ return\ 1$

$else\ return\ 0$